6. Jose

Program Name: Jose.java Input File: jose.dat

Vehicle Identification Numbers or VIN numbers are the unique "name" for each motor vehicle manufactured in the world. A VIN is made up of 17 digits or characters (except for O, I and Q). Each digit or character or group of digits and characters in the VIN has a special meaning. These include the manufacturer, year made, country made and several other features about the vehicle the VIN has been assigned to. The 9th position in the VIN is a check digit in the United States and Canada. That 9th digit or character is where the story of Jose begins.

Jose has just gone to work for the National Highway Traffic Safety Administration which is in charge of VINs in the United States. As the junior programmer in the group, Jose has been tasked with the job of generating the check digit for each of several million potential VINs.

The check digit is generated using this formula. Multiply the value of each digit or character times the weight of the position that digit or character occupies. Find the sum of the products then find the remainder when the sum is divided by eleven. If the remainder is 0 – 9 then that becomes the check digit. If the remainder is 10 then the check digit is X. The check digit is placed in the 9th position in the VIN.

The value of each digit is the digit itself i.e. the value of 5 is 5. The value of each allowable character is shown in this table :

A: 1	B: 2	C: 3	D: 4	E: 5	F: 6	G: 7	H: 8	
J: 1	K: 2	L: 3	M: 4	N: 5		P: 7		R: 9
	S: 2	T: 3	U: 4	V: 5	W: 6	X: 7	Y: 8	Z: 9

The gaps are for the characters that are not allowed and the last line is shifted to the right to preserve the balance of the table.

The weight of each position is shown in this table.

Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Weight	8	7	6	5	4	3	2	10	0	9	8	7	6	5	4	3	2

For example the weight of position 3 is 6 and the weight of position 14 is 5. The weight of position 9 is 0 to ensure that the check digit itself is not part of the sum of the products.

The check digit for 1C3CDFDH8ED721434 (which is 8) is calculated like this.

character	1	C	3	C	D	F	D	H		E	D	7	2	1	4	3	4
value	1	3	3	3	4	6	4	8		5	4	7	2	1	4	3	4
weight	8	7	6	5	4	3	2	10	0	9	8	7	6	5	4	3	2
product	8	21	18	15	16	18	8	80		45	32	49	12	5	16	9	8

8 + 21 + 18 + 15 + 16 + 18 + 8 + 80 + 45 + 32 + 49 + 12 + 5 + 16 + 9 + 8 = 360

360 / 11 is 32 with 8 left over. So, the check digit is 8.

Jose has a file full of incomplete VINs. Each one is missing the check digit. Jose's job is to calculate a check digit for each VIN and place it in the 9th position of the incomplete VIN and print each completed VIN. Of course, Jose is going to write a program to do the job. What would that program look like?

Input: The input file is composed of three sections, one for the table of values assigned to each allowable letter, another for the table of weights assigned to each position and finally a list of incomplete VINs. The first table will contain a line for each allowable letter where the letter is followed by a space and the value. The second table will have a line for each position in the VIN where each line contains the position number followed by a space and then the weight of that position. Both tables will be followed by one blank line. The tables will be followed by a number N that represents the number of incomplete VINs to be processed. The list of incomplete VINs will begin on the next line after N and each will be on a separate line. The 9th position of each VIN will be an underscore.

Output: A list of completed VINs each on a separate line.

(continued next page)

Jose - continued

Sample input: *(Shown in two columns)*

```
A 1
B 2
C 3
D 4
E 5
F 6
G 7
H 8
J 1
K 2
L 3
M 4
N 5
P 7
R 9
S 2
T 3
U 4
V 5
W 6
X 7
Y 8
Z 9

1 8
2 7
3 6
4 5
5 4
6 3
7 2
8 10
9 0
10 9
11 8
12 7
13 6
14 5
15 4
16 3
17 2

10
1C3CDFDH_ED721434
1N4AL3AP_DC290218
1FMJU1HT_FEF20545
1GNDU03E_YD370482
2T3JF4DV_BW140581
3C63R3KJ_EG192714
5TDBT48A_1S268358
5NPDH4AE_DH252374
1LNHL9DR_AG613620
JAACR16E_J7234004
```

Sample output:

```
1C3CDFDH8ED721434
1N4AL3AP1DC290218
1FMJU1HT4FEF20545
1GNDU03E5YD370482
2T3JF4DV0BW140581
3C63R3KJ3EG192714
5TDBT48A61S268358
5NPDH4AEXDH252374
1LNHL9DR9AG613620
JAACR16E9J7234004
```